

# **DBMS PROJECT**

## **ONLINE BANKING SYSTEM**



**Aaryan Raj Saxena [2020004]**

**Prakhar Rai [2020099]**

**Rahul Agrawal [2020108]**

**Sanyam Goyal [2020116]**

### **Problem Statement**

- ❖ To design a database and an application (part of the final evaluation) for an online banking system (like HDFC).

# Project Scope

## ADMIN FEATURES (Employee or Branch Manager)

- ❖ An employee is able to open new and verify different types of accounts of the users.
- ❖ An employee is able to add/remove money to accounts as a teller.
- ❖ An employee can start/end FD/RD for a user as per request.
- ❖ An employee can start and approve the loan applications of the user.
- ❖ An employee can apply/block a credit or debit card at the user's request.
- ❖ An employee can transfer money from one account to another on request.
- ❖ An employee can see transaction logs to check the status of a transaction.
- ❖ An employee can authorize an advertisement for the bank.
- ❖ An employee can verify the documents submitted by the customers for accounting purposes.
- ❖ An employee can issue a passbook at the user's request.

## USER FEATURES (Individual/Joint/Business)

- ❖ A customer can apply for a new account by providing personal information.
- ❖ A customer can view the balance of his/her account.
- ❖ A customer can start/end an FD/RD from the portal whose money will be debited from the account.
- ❖ A customer can apply for personal/collateral loan from the portal.
- ❖ A customer can apply/block for a new credit card/ debit card from the portal.
- ❖ A customer can make bill payments and transfer money from one account to another (NEFT/RTGS/IMPS).
- ❖ A customer can apply for various types of insurance.
- ❖ A customer can make a passbook/checkbook request to a branch for his/her account.

## Assumptions

- ❖ Our online banking system is a closed system where no transactions will happen with any other bank or with any other third party outside the scope of it.
- ❖ Investment-related accounts, plans, etc. things will be considered to be handled by some other banking service providers.
- ❖ Any document/photo in the database will be either stored in form of a file with a character string of some size in the respective field or by providing a link to it placed on the cloud.
- ❖ Any other banking-related services that are not listed on the ER diagram are considered to be managed by some other service providers, therefore, outside the scope of this project.

## Stakeholders

Businesses, employees, customers, financial markets, and government.

## Entities

**Branch** (IFSC\_Code, Branch\_Name, Street, City\_Village\_Town, District, State, Pincode, Opening\_Date)

**Bank\_Account** (Account\_Number, Date\_Of\_Opening, Min\_Balance\_Required, Account\_Balance, Interest\_Rate, Withdrawal\_Limit)

**Department** (Department\_ID, Department\_Name)

**Employee** (Employee\_ID, PAN\_Number, Aadhaar\_Number, First\_Name, Last\_Name, Apartment, Street, City\_Village\_Town, District, State, Pincode, DOB, Age(*Derived Attribute*), Employee\_Phone\_Number, Photo)

**Customer** (CIN, PAN\_Number, First\_Name, Last\_Name, Building, Street, City\_Village\_Town, District, State, {Phone\_Number}, Pincode)

**Collateral** (Collateral\_ID, Collateral\_Name, Collateral\_Value, {Documents\_Link})

**Card** (Card\_Number, Linked\_Mobile\_Number, Card\_Pin, Issued\_Date, Expiry\_Date, CVV, Daily\_Limit, Monthly\_Limit)

**Loan** (Loan\_ID, Loan\_Amount, Interest\_Rate, Tenure\_Months, Loan\_Issued\_Date, Loan\_End\_Date(*Derived Attribute*), Loan\_Given\_To(*FK references Customer(CIN)*), EMI\_Amount)

**Insurance** (Policy\_ID, Policy\_Start\_Date, Policy\_Duration\_Years, Policy\_Expiry\_Date(*Derived Attribute*), Premium\_Amount, Sum\_Insured, Unclaimed\_Amount)

**Asset\_Insurance** (Policy\_ID(*FK references Insurance(Policy\_ID)*), Asset\_Document, Asset\_Value)

**Passbook** (Barcode, Date\_Issued, Date\_Last\_Printed)

**UPI** (UPI\_ID, Active\_Status, Last\_Used, Transaction\_Limit\_Remaining)

**UPI\_Transactions** (UPI\_Transaction\_ID, From\_UPI\_ID(*FK references UPI(UPI\_ID)*), To\_UPI\_ID(*FK references UPI(UPI\_ID)*), Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount)

**Bill\_Payment** (Bill\_ID)

**Auto\_Payment** (Auto\_Payment\_ID, Payment\_Date, Payment\_Frequency)

**Auto\_Payment\_Loan** (Auto\_Payment\_ID(*references Auto\_Payment(Auto\_Payment\_ID)*))

**Auto\_Bill\_Payment** (Auto\_Payment\_ID(*references Auto\_Payment(Auto\_Payment\_ID)*))

**Transaction\_Log** (Log\_ID, Log\_Date\_Time, Log\_File)

**Ad\_Channel** (Channel\_ID, Chanel\_Name, Channel\_Link)

**Advertisement** (Advertisement\_ID, Total\_Cost, Ad\_Type)

**User\_Credentials** (User\_Login\_ID, User\_Login\_Password)

**Employee\_Credentials** (Employee\_Login\_ID, Employee\_Login\_Password)

**Deposit** (Account\_Number(*FK references Bank\_Account(Account\_Number)*), Start\_Date, Duration, Maturity\_Date(*Derived Attribute*), CIN(*FK references Personal\_Customer(CIN)*), Interest\_Rate)

**Chequebook** (Account\_Number(*FK references Bank\_Account(Account\_Number)*), MICR\_Code, Starting\_Cheque\_Number, Pages, Date\_Issued)

**Bank\_Statement** (Account\_Number(*FK references Bank\_Account(Account\_Number)*), Info\_From\_Date, Info\_Till\_Date, Date\_Issued)

**OTPs** (Account\_Number(*FK references Bank\_Account(Account\_Number)*), CIN(*FK references Customer(CIN)*), Valid\_From, Valid\_Till, OTP)

**Installment** (Installment\_ID, Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount, Transaction\_Type)

## Relationships with Attributes

**Branch\_Account** (IFSC\_Code(FK references Branch(IFSC\_Code)), Branch\_Account\_Number(FK references Bank\_Account(Account\_Number)))

**Account\_In\_Branch** (Branch\_Account\_Number(FK references Bank\_Account(Account\_Number)), IFSC\_Code(FK references Branch(IFSC\_Code)))

**Employee\_Account** (Employee\_Account\_Number(FK references Bank\_Account(Account\_Number)), Employee\_ID(references Employee(Employee\_ID)))

**Employee\_In\_Department** (Employee\_ID(references Employee(Employee\_ID)), Department(FK references Department(Department\_ID)))

**Employee\_Works\_Branch** (Employee\_ID(references Employee(Employee\_ID)), IFSC\_Code(FK references Branch(IFSC\_Code)))

**Branch\_Managed\_By** (IFSC\_Code(FK references Branch(IFSC\_Code)), Employee\_ID(FK references Employee(Employee\_ID)), Manages\_From, Manages\_Till)

**Collateral\_Owned\_By** (Collateral\_ID(FK references Collateral(Collateral\_ID)), Collateral\_Owner(FK references Customer(CIN)))

**Personal\_Customer** (CIN(FK references Customer(CIN)), Aadhaar\_Number, DOB, Age(Derived Attribute), Photo)

**Business\_Customer** (CIN(FK references Customer(CIN)), GST, Business\_Name, Business\_Logo)

**Savings\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK references Personal\_Customer(CIN)))

**Current\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK references Business\_Customer(CIN)), Maintenance\_Charges)

**Minor\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), Minor\_First\_Name, Minor\_Last\_Name, Parent\_Account\_Number(FK references Savings\_Account(Account\_Number)), Parent\_CIN(FK references Personal\_Customer(CIN)))

**Zero\_Balance\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK references Personal\_Customer(CIN)))

**Debit\_Card** (Card\_Number(FK references Card(Card\_Number)), Linked\_Account(FK references Bank\_Account(Account\_Number)))

**Prepaid\_Card** (Card\_Number(FK references Card(Card\_Number)), CIN(FK references Personal\_Customer(CIN)), Prepaid\_Balance)

**ATM\_Card** (Card\_Number(FK references Card(Card\_Number)), Linked\_Account(FK references Bank\_Account(Account\_Number)), Cash-Withdrawal\_Limit)

**Credit\_Card** (Card\_Number(FK references Card(Card\_Number)), CIN(FK references Customer(CIN)), Billing\_Date, Due\_Date(Derived Attribute), Credit\_Limit, Total\_Amount\_Due, Rewards\_Point, Minimum\_Amount\_Due(Derived Attribute))

**Account\_Linked\_Credit\_Card** (Card\_Number(FK references CreditCard(Card\_Number)), Linked\_Account(FK references Bank\_Account(Account\_Number)))

**Collateral\_Loan** (Loan\_ID(FK references Loan(Loan\_ID)), Collateral\_ID(FK references Collateral(Collateral\_ID)), Loan\_Type)

**Non\_Collateral\_Loan** (Loan\_ID(FK references Loan(Loan\_ID)), Loan\_Type, {Documents\_PDF})

**Loan\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), Loan\_ID(FK references Loan(Loan\_ID)))

**Insurance\_Given\_To** (Policy\_ID, Nominee\_ID(FK references Customer(CIN)))

**Loan\_Protection\_Insurance** (Policy\_ID(FK references Insurance(Policy\_ID)), Loan\_ID(FK references Loan(Loan\_ID)))

**Term\_Life\_Insurance** (Policy\_ID(FK references Insurance(Policy\_ID)), Insured\_Person(FK references Personal\_Customer(CIN)))

**Medical\_Insurance** (Policy\_ID(FK references Insurance(Policy\_ID)), Insured\_Person(FK references Personal\_Customer(CIN)))

**Transaction\_Details** (Transaction\_ID, From\_Account\_Number(FK references Bank\_Account(Account\_Number)), To\_Account\_Number(FK references Bank\_Account(Account\_Number)), Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount, Transaction\_Type)

**Passbook\_Account** (Barcode(references Passbook(Barcode)), Passbook\_Account\_Number(FK references Bank\_Account(Account\_Number)))

**UPI\_Account** (UPI\_ID(references UPI(UPI\_ID)), Account\_Number\_Linked(FK references Bank\_Account(Account\_Number)))

**Bill\_Payment\_Transactions** (Bill\_ID(references Bill\_Payment(Bill\_ID)), Transaction\_ID(references Transaction\_Details(Transaction\_ID)))

**Card\_Transactions** (Transaction\_ID(references Transaction\_Details(Transaction\_ID)), From\_Card\_Number(FK references Card(Card\_Number)), To\_Card\_Number(FK references Card(Card\_Number)), Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount)

**Auto\_Debit\_Account** (Auto\_Payment\_ID(references Auto\_Payment(Auto\_Payment\_ID), Autodebit\_Account\_Number(FK references Bank\_Account(Account\_Number)))

**Auto\_Loan\_Account** (Auto\_Payment\_ID(references Auto\_Payment\_Loan(Auto\_Payment\_ID), Linked\_Loan\_Account(FK references Loan\_Account(Account\_Number)))

**Auto\_Bill\_To** (Auto\_Payment\_ID(references Auto\_Bill\_Payment(Auto\_Payment\_ID), Bill\_ID(references Bill\_payment(Bill\_ID)))

**Transaction\_With\_Log** (Log\_ID(references Transaction\_Log(Log\_ID)), Transaction\_ID(references Transaction\_Details(Transaction\_ID)))

**Ad\_Channel\_By** (Advertisement\_ID(references Advertisement(Advertisement\_ID)), Channel\_ID(references Ad\_Channel(Channel\_ID)), Authorized\_By(references Employee(Employee\_ID)))

**Credentials\_Of\_User** (User\_Login\_ID(references User\_Credentials(User\_Login\_ID)), User\_ID(references Customer(CIN)))

**Credentials\_Of\_Employee** (Employee\_Login\_ID(references Employee\_Credentials(Employee\_Login\_ID)), Employee\_ID(references Employee(Employee\_ID)))

**Fixed\_Deposit** (Fixed\_Amount)

**Recurring\_Deposit** (Monthly\_Deposit\_Amount)

**Installment\_Of\_Loan** (Loan\_ID(FK references Loan(Loan\_ID)), Installment\_ID(FK references Installment(Installment\_ID)))

# Relational Schemas

**Database:** Online\_Banking\_System

**Branch** (IFSC\_Code, Branch\_Name, Branch\_Account\_Number(*FK references Bank\_Account(Account\_Number)*), Street, City\_Village\_Town, District, State, Pincode, Opening\_Date)

**Bank\_Account** (Account\_Number, Date\_Of\_Opening, Min\_Balance\_Required, Account\_Balance, IFSC\_Code(*FK references Branch(IFSC\_Code)*), Interest\_Rate, Withdrawal\_Limit)

**Department** (Department\_ID, Department\_Name)

**Employee** (Employee\_ID, PAN\_Number, Aadhaar\_Number, First\_Name, Last\_Name, Employee\_Account\_Number(*FK references Bank\_Account(Account\_Number)*), Department(*FK references Department(Department\_ID)*), Apartment, Street, City\_Village\_Town, District, State, Pincode, DOB, Age(*Derived Attribute*), Employee\_Phone\_Number, IFSC\_Code(*FK references Branch(IFSC\_Code)*), Photo)

**Branch\_Managed\_By** (IFSC\_Code(*FK references Branch(IFSC\_Code)*), Employee\_ID(*FK references Employee(Employee\_ID)*), Manages\_From, Manages\_Till)

**Customer** (CIN, PAN\_Number, First\_Name, Last\_Name, Building, Street, City\_Village\_Town, District, State, Pincode)

**Customer\_Phone\_Number** (CIN(*FK references Customer(CIN)*), Phone\_Number)

**Collateral** (Collateral\_ID, Collateral\_Name, Collateral\_Value, Collateral\_Owner(*FK references Customer(CIN)*))

**Documents\_Link** (Document\_ID, Collateral\_ID(*FK references Collateral(Collateral\_ID)*), Document\_Link)

**Personal\_Customer** (CIN(*FK references Customer(CIN)*), Aadhaar\_Number, DOB, Age(*Derived Attribute*), Photo)

**Business\_Customer** (CIN(*FK references Customer(CIN)*), GST, Business\_Name, Business\_Logo)

**Savings\_Account** (Account\_Number(*FK references Bank\_Account(Account\_Number)*), CIN(*FK references Personal\_Customer(CIN)*))



**Current\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK references Business\_Customer(CIN)), Maintenance\_Charges)

**Minor\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), Minor\_First\_Name, Minor\_Last\_Name, Parent\_Account\_Number(FK references Savings\_Account(Account\_Number)), Parent\_CIN(FK references Personal\_Customer(CIN)))

**Zero\_Balance\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK references Personal\_Customer(CIN)))

**Card** (Card\_Number, Linked\_Mobile\_Number, Card\_Pin, Issued\_Date, Expiry\_Date, CVV, Daily\_Limit, Monthly\_Limit)

**Debit\_Card** (Card\_Number(FK references Card(Card\_Number)), Linked\_Account(FK references Bank\_Account(Account\_Number)))

**Prepaid\_Card** (Card\_Number(FK references Card(Card\_Number)), CIN(FK references Personal\_Customer(CIN)), Prepaid\_Balance)

**ATM\_Card** (Card\_Number(FK references Card(Card\_Number)), Linked\_Account(FK references Bank\_Account(Account\_Number)), Cash-Withdrawal\_Limit)

**Credit\_Card** (Card\_Number(FK references Card(Card\_Number)), CIN(FK references Customer(CIN)), Billing\_Date, Due\_Date(Derived Attribute), Credit\_Limit, Total\_Amount\_Due, Rewards\_Point, Minimum\_Amount\_Due(Derived Attribute))

**Account\_Linked\_Credit\_Card** (Card\_Number(FK references CreditCard(Card\_Number)), Linked\_Account(FK references Bank\_Account(Account\_Number)))

**Loan** (Loan\_ID, Loan\_Amount, Interest\_Rate, Tenure\_Months, Loan\_Issued\_Date, Loan\_End\_Date(Derived Attribute), Loan\_Given\_To(FK references Customer(CIN)), EMI\_Amount)

**Collateral\_Loan** (Loan\_ID(FK references Loan(Loan\_ID)), Collateral\_ID(FK references Collateral(Collateral\_ID)), Loan\_Type)

**Non\_Collateral\_Loan** (Loan\_ID(FK references Loan(Loan\_ID)), Loan\_Type)

**Documents\_PDF** (Document\_ID, Loan\_ID(FK references Non\_Collateral\_Loan(Loan\_ID)), Document\_PDF)

**Loan\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)), Loan\_ID(FK references Loan(Loan\_ID)))

**Insurance** (Policy\_ID, Policy\_Start\_Date, Policy\_Duration\_Years, Policy\_Expiry\_Date(*Derived Attribute*), Premium\_Amount, Sum\_Insured, Unclaimed\_Amount, Nominee\_ID(*FK references Customer(CIN)*))

**Asset\_Insurance** (Policy\_ID(*FK references Insurance(Policy\_ID)*), Asset\_Document, Asset\_Value)

**Loan\_Protection\_Insurance** (Policy\_ID(*FK references Insurance(Policy\_ID)*), Loan\_ID(*FK references Loan(Loan\_ID)*))

**Term\_Life\_Insurance** (Policy\_ID(*FK references Insurance(Policy\_ID)*), Insured\_Person(*FK references Personal\_Customer(CIN)*))

**Medical\_Insurance** (Policy\_ID(*FK references Insurance(Policy\_ID)*), Insured\_Person(*FK references Personal\_Customer(CIN)*))

**Transaction\_Details** (Transaction\_ID, From\_Account\_Number(*FK references Bank\_Account(Account\_Number)*), To\_Account\_Number(*FK references Bank\_Account(Account\_Number)*), Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount, Transaction\_Type)

**Passbook** (Barcode, Passbook\_Account\_Number(*FK references Bank\_Account(Account\_Number)*), Date\_Issued, Date\_Last\_Printed)

**UPI** (UPI\_ID, Account\_Number\_Linked(*FK references Bank\_Account(Account\_Number)*), Active\_Status, Last\_Used, Transaction\_Limit\_Remaining)

**UPI\_Transactions** (UPI\_Transaction\_ID, From\_UPI\_ID(*FK references UPI(UPI\_ID)*), To\_UPI\_ID(*FK references UPI(UPI\_ID)*), Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount)

**Bill\_Payment** (Bill\_ID, Transaction\_ID(*references Transaction\_Details(Transaction\_ID)*))

**Card\_Transactions** (Transaction\_ID(*references Transaction\_Details(Transaction\_ID)*), From\_Card\_Number(*FK references Card(Card\_Number)*), To\_Card\_Number(*FK references Card(Card\_Number)*), Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount)

**Auto\_Payment** (Auto\_Payment\_ID, Payment\_Date, Payment\_Frequency, Autodebit\_Account\_Number(*FK references Bank\_Account(Account\_Number)*))

**Auto\_Payment\_Loan** (Auto\_Payment\_ID(*references Auto\_Payment(Auto\_Payment\_ID)*), Linked\_Loan\_Account(*FK references Loan\_Account(Account\_Number)*))

**Auto\_Bill\_Payment** (Auto\_Payment\_ID(references Auto\_Payment(Auto\_Payment\_ID)),  
Bill\_ID(references Bill\_payment(Bill\_ID)))

**Transaction\_Log** (Log\_ID, Log\_Date\_Time, Transaction\_ID(references  
Transaction\_Details(Transaction\_ID)), Log\_File)

**Ad\_Channel** (Channel\_ID, Chanel\_Name, Channel\_Link)

**Advertisement** (Advertisement\_ID, Total\_Cost, Ad\_Type, Channel\_ID(references  
Ad\_Channel(Channel\_ID)), Authorized\_By(references Employee(Employee\_ID)))

**User\_Credentials** (User\_Login\_ID, User\_Login\_Password, User\_ID(references Customer(CIN)))

**Closed\_Account** (Account\_Number(FK references Bank\_Account(Account\_Number)),  
Date\_of\_Closing)

**Blocked\_Card** (Card\_Number(FK references Card(Card\_Number)), Date\_of\_Blocking)

**Employee\_Credentials** (Employee\_Login\_ID, Employee\_Login\_Password,  
Employee\_ID(references Employee(Employee\_ID)))

**Fixed\_Deposit** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK  
references Personal\_Customer(CIN)), Fixed\_Amount, Start\_Date, Duration, Maturity\_Date(Derived  
Attribute), Interest\_Rate)

**Recurring\_Deposit** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK  
references Personal\_Customer(CIN)), Monthly\_Deposit\_Amount, Start\_Date, Duration,  
Maturity\_Date(Derived Attribute), Interest\_Rate)

**Chequebook** (Account\_Number(FK references Bank\_Account(Account\_Number)),  
MICR\_Code(Derived Attribute), Starting\_Cheque\_Number, Pages, Date\_Issued)

**Bank\_Statement** (Account\_Number(FK references Bank\_Account(Account\_Number)),  
Info\_From\_Date, Info\_Till\_Date, Date\_Issued)

**OTPs** (Account\_Number(FK references Bank\_Account(Account\_Number)), CIN(FK references  
Customer(CIN)), Valid\_From, Valid\_Till, OTP)

**Installment** (Loan\_ID(FK references Loan(Loan\_ID)), Transaction\_Date\_Time,  
Transaction\_Status, Transaction\_Amount, Transaction\_Type)

## Weak Entities

- ❖ **Transaction\_Details** (Transaction\_ID, From\_Account\_Number(FK references *Bank\_Account*(*Account\_Number*)), To\_Account\_Number(FK references *Bank\_Account*(*Account\_Number*)), *Transaction\_Date\_Time*, *Transaction\_Status*, *Transaction\_Amount*, *Transaction\_Type*)
  - Dependent on the **Bank\_Account** entity since transactions happen between the accounts.
- ❖ **UPI\_Transactions** (UPI\_Transaction\_ID, From\_UPI\_ID(FK references *UPI*(*UPI\_ID*)), To\_UPI\_ID(FK references *UPI*(*UPI\_ID*)), *Transaction\_Date\_Time*, *Transaction\_Status*, *Transaction\_Amount*)
  - Dependent on the **UPI** entity since UPI transactions happen from one UPI id to another.
- ❖ **OTPs** (Account\_Number(FK references *Bank\_Account*(*Account\_Number*)), *CIN*(FK references *Customer*(*CIN*)), Valid\_From, Valid\_Till, *OTP*)
  - Dependent on the **Bank\_Account** entity since OTPs are sent to an authorized customer on a particular account only.
- ❖ **Fixed\_Deposit** (Account\_Number(FK references *Bank\_Account*(*Account\_Number*)), *CIN*(FK references *Personal\_Customer*(*CIN*)), *Fixed\_Amount*, *Start\_Date*, *Duration*, *Maturity\_Date*(Derived Attribute), *Interest\_Rate*)
  - Dependent on the **Bank\_Account** entity since deposits are made to an account only.
- ❖ **Recurring\_Deposit** (Account\_Number(FK references *Bank\_Account*(*Account\_Number*)), *CIN*(FK references *Personal\_Customer*(*CIN*)), *Monthly\_Deposit\_Amount*, *Start\_Date*, *Duration*, *Maturity\_Date*(Derived Attribute), *Interest\_Rate*)
  - Dependent on the **Bank\_Account** entity since deposits are made to an account only.

- ❖ **Chequebook** (Account\_Number(FK references Bank\_Account(Account\_Number)), MICR\_Code(*Derived Attribute*), Starting\_Cheque\_Number, Pages, Date\_Issued)
  - *Dependent on the **Bank\_Account** entity since chequebooks are issued to an account only.*
- ❖ **Bank\_Statement** (Account\_Number(FK references Bank\_Account(Account\_Number)), Info\_From\_Date, Info\_Till\_Date, Date\_Issued)
  - *Dependent on the **Bank\_Account** entity since monthly reports in the form of bank statements are for an account only.*
- ❖ **Installment** (Loan\_ID(FK references Loan(Loan\_ID)), Transaction\_Date\_Time, Transaction\_Status, Transaction\_Amount, Transaction\_Type)
  - *Dependent on the **Loan** entity since installments are made to a particular loan only.*

## Ternary Relationships

- ❖ **Ad\_Channel\_By** (Advertisement\_ID(references Advertisement(Advertisement\_ID)), Channel\_ID(references Ad\_Channel(Channel\_ID)), Authorized\_By(references Employee(Employee\_ID)))
  - *Relationship between **Advertisement**, **Ad\_Channel**, and **Employee** since each ad is released on only one channel and authorized by only one employee*

## SQL Queries

**USE online\_banking\_system;**

**-- Query 1 [select, order by asc/desc]**

```
SELECT IFSC_Code, Branch_Name, District, State
FROM Branch
ORDER BY State ASC, Branch_Name DESC;
```

**-- Query 2 [insert]**

```
INSERT INTO Branch_Managed_By
VALUES (100014, 10026, '2027-11-12', '2030-11-12');
SELECT * FROM Branch_Managed_By;
-- DELETE FROM Branch_managed_By WHERE Employee_Id = 10026;
```

**-- Query 3 [update]**

```
UPDATE Employee
SET Department = 1
WHERE Employee_ID = 10026;
SELECT * FROM Employee;
```

**-- Query 4 [delete]**

```
DELETE FROM Customer_Phone_Number
WHERE CIN = 100000400;
-- INSERT INTO Customer_Phone_Number
-- VALUES (100000400, 6679930452);
SELECT * FROM Customer_Phone_Number;
```

**-- Query 5 [sum, if\_null]**

```
SELECT COALESCE(SUM(Collateral_Value), 0)
AS Total_Collateral_Value
FROM Collateral;
```

**-- Query 6 [case when then else]**

```
SELECT Account_Number,
CASE
WHEN Account_Balance > 500000 THEN 'The balance is greater than 5 lakhs'
```

```
WHEN Account_Balance > 100000 THEN 'The balance is between 1 to 5 lakhs'  
ELSE 'The balance is under 1 lakh'  
END AS Balance  
FROM Bank_Account;
```

**-- Query 7 [exists]**

```
SELECT Card_Number, Linked_Mobile_Number  
FROM Card  
WHERE EXISTS (SELECT * FROM Debit_Card WHERE Debit_Card.Card_Number =  
Card.Card_Number);
```

**-- Query 8 [like]**

```
SELECT * FROM Customer  
WHERE City_Village_Town LIKE '%_pur';
```

**-- Query 9 [between, and]**

```
SELECT * FROM Transaction_Details  
WHERE Transaction_Status = 'success' AND Transaction_Date_Time BETWEEN  
'2022-01-01 00:00:00' AND '2022-02-01 00:00:00';
```

**-- Query 10 [count, group by, having]**

```
SELECT COUNT(Account_Number) , Min_Balance_Required  
FROM Bank_Account  
GROUP BY Min_Balance_Required  
Having COUNT(Account_Number)>100;
```

**-- Query 11 [inner join all]**

```
SELECT Account_Number, Card_Number, Cash-Withdrawl_Limit  
FROM Bank_Account INNER JOIN ATM_Card ON Account_Number = Linked_Account;
```

**-- Query 12 [left join]**

```
SELECT Account_Number, Card_Number, Cash-Withdrawl_Limit  
FROM Bank_Account LEFT JOIN ATM_Card ON Account_Number = Linked_Account;
```

**-- Query 13 [inner join some]**

```
SELECT Account_Number, Card_Number, Cash-Withdrawl_Limit
FROM Bank_Account INNER JOIN ATM_Card ON Account_Number = Linked_Account
WHERE Account_Number = '11000000120';
```

**-- Query 14 [union names]**

```
SELECT first_name FROM Customer
UNION
SELECT first_name FROM Employee
ORDER BY first_name;
```

**-- Query 15 [alter table]**

```
ALTER TABLE Employee
ADD CONSTRAINT chk_age CHECK (Age>=18);
-- ALTER TABLE Employee
-- DROP CONSTRAINT chk_age;
SELECT * FROM Employee;
```

**-- Query 16 [create view]**

```
CREATE VIEW JalandharCustomer AS
SELECT *
FROM Customer
WHERE City_Village_Town = 'Jalandhar';
SELECT * FROM JalandharCustomer;
```

**-- Query 17 [authorization]**

```
CREATE USER IF NOT EXISTS 'ars'@'localhost'
IDENTIFIED BY 'ars';
GRANT INSERT
ON online_banking_system.advertisement
TO 'ars'@'localhost';
SHOW GRANTS FOR 'ars'@'localhost';
REVOKE INSERT
ON online_banking_system.advertisement
FROM 'ars'@'localhost';
DROP USER IF EXISTS 'ars'@'localhost';
```



**-- [View 1]**

```
CREATE VIEW JalandharCustomer AS
SELECT * FROM Customer
WHERE City_Village_Town = 'Jalandhar';
```

**-- [View 2]**

```
CREATE VIEW HighDefaultCustomer AS
SELECT * FROM Loan
WHERE loan_amount >= 200000;
```

**-- [View 3]**

```
CREATE VIEW HighUpiTransactions AS
SELECT * FROM upi_transactions
WHERE transaction_amount >= 40000;
```

**-- [View 4]**

```
CREATE VIEW MedicalInsuranceID AS
SELECT medical_insurance.policy_id, insurance.nominee_id,
insurance.policy_start_date, insurance.policy_duration_years, insurance.premium_amount
FROM medical_insurance inner join insurance
ON medical_insurance.policy_id = insurance.policy_id;
```

**-- [Grant 1]**

```
CREATE USER IF NOT EXISTS 'advt_board'@'localhost'
IDENTIFIED BY 'advt_board';
GRANT SELECT, INSERT, UPDATE, DELETE
ON advertisement
TO 'advt_board'@'localhost';
SHOW GRANTS FOR 'advt_board'@'localhost';
-- REVOKE INSERT
-- ON advertisement
-- FROM 'advt_board'@'localhost';
-- DROP USER IF EXISTS 'advt_board'@'localhost';
```

**-- [Grant 2]**

```
CREATE USER IF NOT EXISTS 'collateral_dept'@'localhost'  
IDENTIFIED BY 'collateral_dept';  
GRANT SELECT, INSERT, UPDATE, DELETE  
ON collateral  
TO 'collateral_dept'@'localhost';  
SHOW GRANTS FOR 'collateral_dept'@'localhost';  
-- REVOKE INSERT  
-- ON collateral  
-- FROM 'collateral_dept'@'localhost';  
-- DROP USER IF EXISTS 'collateral_dept'@'localhost';
```

**-- [Index 1] -- customer->pincode**

```
CREATE INDEX `idx_customer_pincode` ON  
`online_banking_system`.`customer` (pincode)  
COMMENT " ALGORITHM DEFAULT LOCK DEFAULT;
```

**-- [Index 2] -- employee->department**

```
CREATE INDEX `idx_employee_department` ON  
`online_banking_system`.`employee` (department)  
COMMENT " ALGORITHM DEFAULT LOCK DEFAULT;
```

**-- [Index 3] -- bank\_account->account\_balance**

```
CREATE INDEX `idx_bank_account_account_balance` ON  
`online_banking_system`.`bank_account` (account_balance)  
COMMENT " ALGORITHM DEFAULT LOCK DEFAULT;
```

**-- [Index 4] -- loan->loan\_end\_date**

```
CREATE INDEX `idx_loan_loan_end_date` ON  
`online_banking_system`.`loan` (loan_end_date)  
COMMENT " ALGORITHM DEFAULT LOCK DEFAULT;
```

**-- [Index 5] -- upi->upi\_id**

```
CREATE INDEX `idx_upi_upi_id` ON  
`online_banking_system`.`upi` (upi_id)  
COMMENT " ALGORITHM DEFAULT LOCK DEFAULT;
```

-- **[Index 6]** -- insurance->nominee\_id

```
CREATE INDEX `idx_insurance_nominee_id`  
ON `online_banking_system`.`insurance` (nominee_id)  
COMMENT " ALGORITHM DEFAULT LOCK DEFAULT;
```

-- **[Trigger 1]**

```
DROP TRIGGER IF EXISTS `online_banking_system`.`bank_account_BEFORE_INSERT`;  
DELIMITER $$  
USE `online_banking_system`$$  
CREATE DEFINER = CURRENT_USER TRIGGER  
`online_banking_system`.`bank_account_BEFORE_INSERT` BEFORE INSERT  
ON `bank_account` FOR EACH ROW  
BEGIN  
    IF NEW.account_balance < NEW.min_balance_required THEN  
        SET NEW.account_balance = NEW.min_balance_required;  
    END IF;  
END$$  
DELIMITER ;
```

-- **[Trigger 2]**

```
DROP TRIGGER IF EXISTS `online_banking_system`.`card_BEFORE_INSERT`;  
DELIMITER $$  
USE `online_banking_system`$$  
CREATE DEFINER=`root`@`localhost` TRIGGER `card_BEFORE_INSERT` BEFORE INSERT  
ON `card` FOR EACH ROW BEGIN  
    IF NEW.expiry_date < NEW.issued_date THEN  
        SET NEW.expiry_date = NEW.issued_date+1;  
    END IF;  
END$$  
DELIMITER ;
```

**-- [Trigger 3]**

```
DROP TRIGGER IF EXISTS `online_banking_system`.`fixed_deposit_BEFORE_INSERT`;
DELIMITER $$
USE `online_banking_system`$$
CREATE DEFINER = CURRENT_USER TRIGGER
`online_banking_system`.`fixed_deposit_BEFORE_INSERT` BEFORE INSERT
ON `fixed_deposit` FOR EACH ROW
BEGIN
    IF NEW.duration_years = 0 THEN SET NEW.duration_years = 1;
end if;
END$$
DELIMITER ;
```

**-- [Trigger 4]**

```
DROP TRIGGER IF EXISTS
`online_banking_system`.`recurring_deposit_BEFORE_INSERT`;
DELIMITER $$
USE `online_banking_system`$$
CREATE DEFINER = CURRENT_USER TRIGGER
`online_banking_system`.`recurring_deposit_BEFORE_INSERT` BEFORE INSERT
ON `recurring_deposit` FOR EACH ROW
BEGIN
    IF NEW.duration_years = 0 THEN SET NEW.duration_years = 1;
end if;
END$$
DELIMITER ;

SHOW TRIGGERS;
```